

Логический вывод и элементы
планирования действий в группах
роботов

Воробьев В.В.

НИЦ Курчатовский институт

Введение

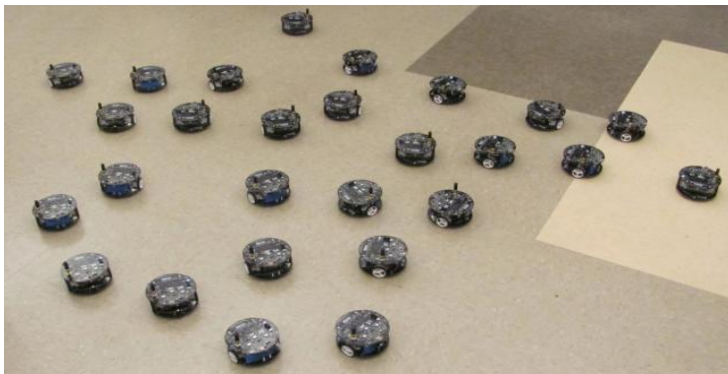
Проблематика робототехники:

- структура особи;
- хранение данных и организация запросов к базе данных;
- распределение ролей;
- организация запросов к БД;
- разрешение конфликтных ситуаций;
- **выработка общего решения (логический вывод);**
- и т.д.

групповой

Способ организации групп роботов – локальный, т.к. ни один из роботов (подсистем) этой группы (системы) не располагает достаточно полной информацией о других роботах (подсистемах).

Частный случай – робот может “общаться” только со своими ближайшими соседями.



[Стефанюк, 2004] Стефанюк В.Л. Локальная организация интеллектуальных систем. -М.: Физматлит, 2004. – 328с.

Постановка задачи

Цель – реализация механизма логического вывода в логике первого порядка в группах роботов с локальным взаимодействием.

Особенность локального взаимодействия в группах роботов – низкоскоростные и малонадежные каналы связи с соседями.

Задачи:

- решение проблемы “старта” логического вывода;
- решение проблемы “останова” логического вывода;
- реализация механизма “кэширования” ;
- реализация механизма ретрансляций запросов-ответов.

“Ошибочные” аналогии:

- группа роботов – вычислительная сеть. Использование стратегий распараллеливания вывода;
- база данных группы роботов – распределенная БД.

Возможное решение поставленных задач

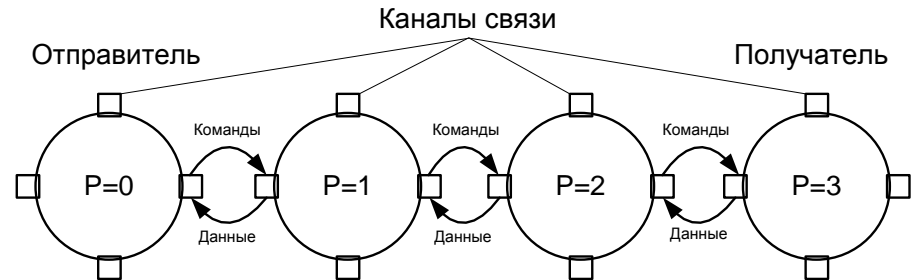
Задача	Решение
Проблема “старта”	Использование механизмов выбора лидера
Проблема доступа к БД остальных роботов группы	Ретрансляция запросов от лидера Ретрансляция ответов к лидеру
Проблема “неупорядоченной” БД	Реализация механизма кэширования при логическом выводе [Russell, Norvig, 1995]
Проблема “останова”	Вывод заканчивается неуспешно если лидеру и его соседям не удалось сопоставить целевое выражение

[Russel, Norvig, 1995] Russell S.J., Norvig P. Artificial Intelligence: A Modern Approach. , 1995. 946 с.

Пример реализации механизмов

Механизм ретрансляций

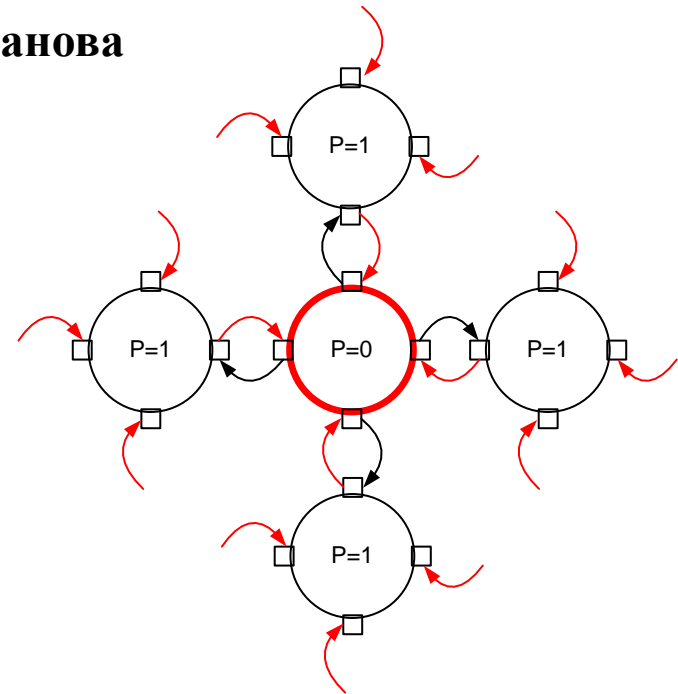
$R_{отп} < R_{пол}$ – принимается запрос, игнорируется ответ
 $R_{отп} > R_{пол}$ – принимается ответ, игнорируется запрос



Механизм останова

Неуспех – сопоставимое выражение не найдено в своей БД и его БД соседей (рекурсивно).

Красная стрелка – ответ о неуспешном выполнении запроса.



Алгоритм логического вывода

Goal – список целевых выражений для доказательства, Base – база фактов и правил робота, N – номер доказываемого выражения из списка Goal

Начало

i = 0

//Запрос соседям выражения Goal на сопоставление

Request(Goal[N])

если V-лидер то

 //Пока не просмотрена вся база данных

цикл пока (i != len(Base))

 //Сопоставление целевого выражения с выражением в базе

если Compare(Goal[N],Base[i]) == true **то**

 //Вывод следующей цели

 Inference(Goal[N+1])

иначе

 i = i+1

иначе

 //Пока не просмотрена вся база данных

цикл пока (i != len(Base))

 //Сравнение целевого выражения с выражением в базе

если Compare(Goal[N],Base[i])==true **то**

 //Отправить результат сопоставления

 Transmit(Result(Goal[N],Base[i]))

иначе

 i = i+1

Конец

Архитектура системы

ROS



Система локализации – определение местоположения роботов в пространстве.

СУ верхнего уровня – осуществляет логический вывод.

Сервер окружения – эмуляция локальной связи роботов.

Интерфейс с пользователем – формирование запроса от пользователя в необходимый формат. Узел не обязателен.

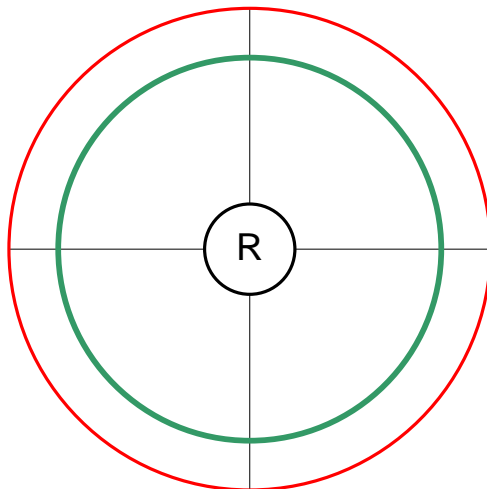
1 – Навигационные данные.

2 – Запрос от пользователя. Принимается только лидером.

3 – Данные, посылаемые роботом по локальной связи.

4 – Данные, принимаемые роботом по локальной связи.

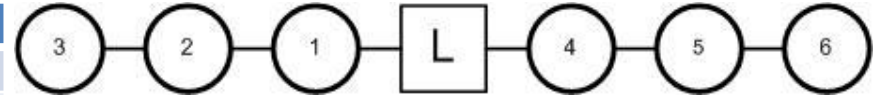
Пересечение внешней окружности – потеря связи, внутренней – установление связи



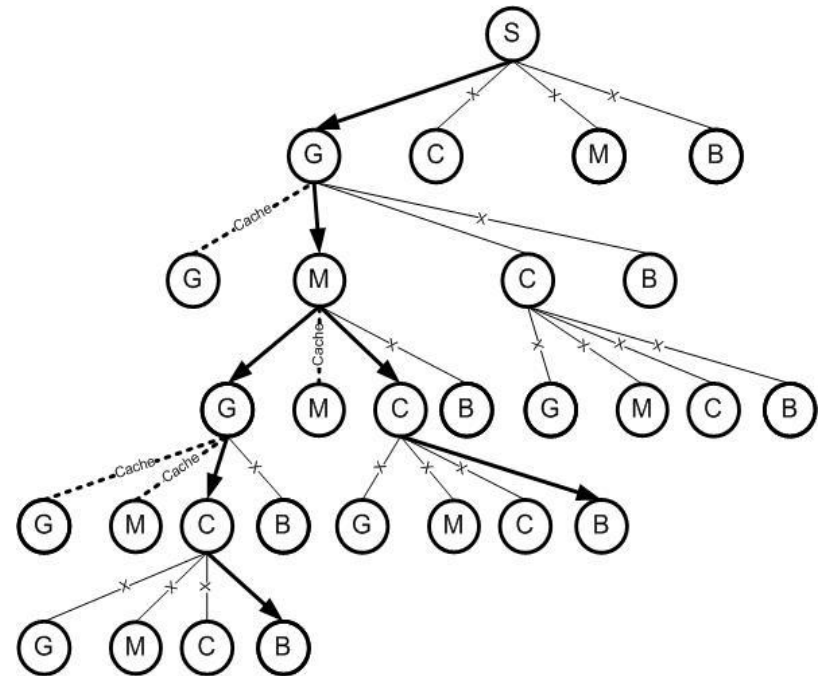
Пример решения задачи

Задача “обезьяна и банан” [Братко, 1990]. Конфигурация группы роботов:

№	Правило или факт
1	can(state(-,-,-,yes))
2	step(state(P1,onfloor,B,H),go(P1,P2),state(P2,onfloor,B,H))
3	step(state(P1,onfloor,P1,H),move(P1,P2),state(P2,onfloor,P2,H))
4	can(S1):step(S1,St,S2),can(S2)
5	step(state(P,onfloor,P,H),climb,state(P,onbox,P,H))
6	step(state(onmiddle,onbox,onmiddle,no),bring,state(onmiddle,onbox,onmiddle,yes))



Дерево поиска:



[Братко, 1990] Братко И. Программирование на языке Пролог для искусственного интеллекта. – Москва: Мир , 1990. 560 с.

Заключение

- Реализован механизм логического вывода для группы роботов с локальным взаимодействием, использующий элементы логики первого порядка.
- Реализован алгоритм кэширования в процессе логического вывода, позволяющий избежать ситуации с бесконечным поиском в существенном количестве случаев.
- Описанный алгоритм может решать ряд задач из области планирования совместных действий в группе роботов.
- Реализованы операторы управления поиском fail и !.